

# PERSONAL COMPUTER SYSTEM

**Patent number:** JP3091838  
**Publication date:** 1991-04-17  
**Inventor:** RICHIAADO BIIIRUKOFUSUKII; JIYON UIREI  
 BURATSUKURETSUJI J; DOIRU SUTANFUIIRU  
 KURONKU; RICHIAADO AREN DAIAN; SUKOTSUTO  
 JIERARUDO KINIAA; JIYOOJI DEII KOBATSUKU;  
 MASHIYUU SUTEIBUN PORUKA JIYUN; ROBAATO  
 SAKUSENMAIAA; KEBIN MAASHIYARU JIBOROSUKII;  
 JIERII DEYUUN DEIKISHION; ANDORIYUU BOISU  
 MAKUNEIRU; EDOWAADO IIBUNINGU  
 WATSUCHITER  
**Applicant:** IBM  
**Classification:**  
 - international: G06F9/445; G06F13/00; G06F13/10  
 - european: G06F9/445B4  
**Application number:** JP19900220132 19900823  
**Priority number(s):** US19890398820 19890825

## Also published as:

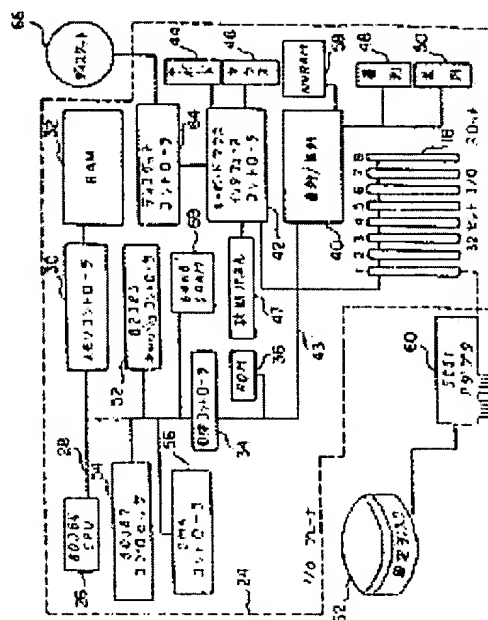
EP0417889 (A2)  
 US5022077 (A1)  
 MX171879 (A)  
 EP0417889 (A3)  
 DE4026912 (A1)

more >>

Report a data error here

## Abstract of JP3091838

**PURPOSE:** To prevent a BIOS code stored in a fixed disk from being changed without permission by providing a system processor, ROM, RAM, direct access storage device, and the same controller. **CONSTITUTION:** The first part of a BIOS is stored in an ROM 36, a direct access storage device 32 stores a master boot record including an executable code segment and the second part of the BIOS in a specific area, and a memory controller 30 protects the specific area, and permits access in response to a reset signal. A processor 26 is initialized by the first part of the BIOS of the ROM 36, and the reset signal is applied to the controller 30, and a CPU 26 performs access to the master boot record in order to set the master boot record loadable to the ROM 32, and moves the control to the executable code segment. The second part of the BIOS is loaded to the RAM 32, the control is moved, and the OS is booted. The second part prevents the access to the specific area by the protecting means of the controller 30 during the normal operation of the OS. Thus, the unauthorized change of the BIOS of a fixed disk 62 with especially large capacity can be prevented.



Data supplied from the esp@cenet database - Worldwide



⑩ 日本国特許庁 (J P)

⑪ 特許出願公開

⑫ 公開特許公報 (A)

平3-91838

⑬ Int. Cl. 5

識別記号

庁内整理番号

⑭ 公開 平成3年(1991)4月17日

G 06 F 9/445  
13/00

3 0 5 J

7629-5B  
7361-5B

G 06 F 9/06 4 2 0 H※

審査請求 有 請求項の数 24 (全22頁)

⑮ 発明の名称 パーソナルコンピュータシステム

⑯ 特 願 平2-220132

⑰ 出 願 平2(1990)8月23日

優先権主張 ⑱ 1989年8月25日 ⑲ 米国 (U S) ⑳ 398820

⑳ 発 明 者 リチャード・ビール アメリカ合衆国フロリダ州デルレイ・ビーチ、ハミングバード・ドライブ1401番地

㉑ 発 明 者 ジョン・ウイレイ・ブラツクレツジ、ジュニア アメリカ合衆国フロリダ州ボカ・ラトン、シイクワイア・レーン304番地

㉒ 出 願 人 インターナショナル・ビジネス・マシーンズ・コーポレーション アメリカ合衆国10504、ニューヨーク州 アーモンク (番地なし)

㉓ 代 理 人 弁理士 頃 宮 孝一 外1名

最終頁に続く

明 細 書

1. 発明の名称 パーソナルコンピュータシステム

2. 特許請求の範囲

(1) オペレーティングシステムを実行するシステムプロセッサと、

B I O S の第1部分を記憶する読み取り専用メモリと、

ランダムアクセスメモリと、

実行可能コードセグメントを含むマスタブートレコード及び前記 B I O S の第2部分を特定の領域に記憶する直接アクセス記憶装置と、

前記特定の領域を保護し、リセット信号にตอบสนองして前記特定の領域へのアクセスを許可する保護手段を有する直接アクセス記憶装置コントローラとを備え、

前記第1部分はシステムプロセッサを初期設定するとともに、前記リセット信号を発生して前記コントローラへ供給することにより、前記システ

ムプロセッサが前記マスタブートレコードをアクセスして前記ランダムアクセスメモリにロードできるようにし、更に制御を前記実行可能コードセグメントに移し、

前記実行可能セグメントは制御が移されたことに応答して、前記第2部分を前記ランダムアクセスメモリにロードし、更に前記オペレーティングシステムをブートするために制御を前記第2部分に移し、

前記第2部分は、前記オペレーティングシステムの通常動作の間、前記特定の領域のアクセスを禁止するために前記保護手段を活動化する、

ことを特徴とするパーソナルコンピュータシステム。

(2) 前記第1部分は、電源投入により、前記リセット信号の発生を開始する、請求項1に記載のシステム。

(3) 前記第1部分は、システムがリセット状態になったことにตอบสนองして、前記リセット信号の発生を開始する、請求項1に記載のシステム。

(4) 前記マスタブートレコードは、該レコードに適合したハードウェア構成を表わすデータセグメントを含み、前記読み取り専用メモリは、前記システムプロセッサのハードウェア構成を表わすデータを含み、前記第1部分は、前記第2部分が前記ランダムアクセスメモリにロードされる前に、前記マスタブートレコード及び前記読み取り専用メモリからのハードウェア構成データを比較して、前記マスタブートレコードが前記システムプロセッサに適合しているかどうかを確かめる、請求項1に記載のシステム。

(5) 前記データセグメントは、前記マスタブートレコードに適合したシステムブレーナを表わす値を含み、システムブレーナは、自身を一意的に識別する手段を有する、請求項4に記載のシステム。

(6) 前記マスタブートレコードのハードウェア構成データは、該マスタブートレコードに適合したシステムプロセッサを識別するモデル値と、該マスタブートレコードに適合したシステムブレーナのI/O構成を表わすサブモデル値とを含み、前

記読み取り専用メモリは、システムプロセッサを識別する対応モデル値及びシステムブレーナのI/O構成を表わす対応サブモデル値を含み、前記モデル値及び前記サブモデル値を前記対応モデル値及び前記対応サブモデル値とそれぞれ比較することにより、前記マスタブートレコードが前記システムプロセッサ及び前記システムブレーナのI/O構成に適合しているかどうかを確かめる、請求項4に記載のシステム。

(7) 前記システムプロセッサと電気的に接続された不揮発性ランダムアクセスメモリを更に備え、該メモリはシステム構成を表わすデータを含み、該データはシステムの構成が変わると更新され、前記第1部分は、該データを前記読み取り専用メモリ中の対応するデータと比較することにより、システムの構成が変わったかどうかを調べる、請求項1に記載のシステム。

(8) 前記システムプロセッサは、順番に番号を付けられたブロックの形で前記コントローラヘッダレコードを転送し、前記マスタブートレコード

及び前記第2部分は番号の大きい方のブロックにある、請求項1に記載のシステム。

(9) 前記保護手段はアドレス可能な最大ブロックを設定し、該最大ブロックは前記マスタブートレコード及び前記第2部分の最も番号の小さいブロックであり、前記保護手段は前記最大ブロック以上の番号のブロックへのアクセスを禁止するとともに、前記最大ブロックより小さい番号のブロックへのアクセスを許可する、請求項8に記載のシステム。

(10) システムプロセッサと、  
主メモリと、

複数のデータレコード、該データレコードを前記主メモリへロードするためのロード手段、および該ロード手段によって前記主メモリに読み込まれる主メモリ常驻プログラムイメージを含み、そのうち前記ロード手段および前記主メモリ常驻プログラムイメージは保護可能な領域に記憶している少なくとも1つの直接アクセス記憶装置と、

前記システムプロセッサを初期設定するととも

に、前記直接アクセス記憶装置から前記主メモリに前記ロード手段を読み込ませる第1プログラムを含む読み取り専用メモリと、

前記主メモリ常驻プログラムイメージが前記主メモリに読み込まれることによつて生成される主メモリ常驻プログラムにより活動化され、前記直接アクセス記憶装置の前記保護可能な領域を保護して、前記ロード手段及び前記主メモリ常驻プログラムイメージが許可なくアクセスされるのを禁止する保護手段と、

を具備するパーソナルコンピュータシステム。

(11) 前記ロード手段は、当該システムが前記主メモリ常驻プログラムに適合しているかどうかを確かめる検査手段を含む、請求項10に記載のシステム。

(12) 前記ロード手段は、前記主メモリ常驻プログラムのローディングを実行するための実行可能コードセグメントを有するマスタブートレコードを含み、前記第1プログラムが前記実行可能コードセグメントに制御を移すことによつて前記主メモリ

への前記主メモリ常駐プログラムイメージのローディングを行わせる、請求項11に記載のシステム。

(13)前記第1プログラムは、前記主メモリ常駐プログラムをロードするのに必要なシステム機能だけを初期設定して検査する電源投入自己検査ルーチンを含む、請求項10に記載のシステム。

(14)前記電源投入自己検査ルーチンは、システムプロセッサ機能、メモリサブシステムおよび直接アクセス記憶装置サブシステムを初期設定する、請求項13に記載のシステム。

(15)前記検査手段は、システムプロセッサのタイプおよびシステムプロセッサに接続されるシステムプレーナの構成を表わすデータを含み、請求項11に記載のシステム。

(16)前記直接アクセス記憶装置は、固定ディスクおよびディスクコントローラを含み、前記システムプロセッサは番号を付されたブロックの形でデータを前記ディスクコントローラへ転送し、前記保護可能な領域には番号の大きい方のブロックが割

当てられる、請求項10に記載のシステム。

(17)前記保護手段は、前記システムプロセッサからのブロック番号と前記保護可能な領域に割当てられたブロックの最小番号を比較し、前記ブロック番号が該最小番号よりも小さい場合にのみアクセスを許可する、請求項16に記載のシステム。

(18)前記第1プログラムは、電源投入またはリセット条件に応答してリセット信号の発生を開始する、請求項10に記載のシステム。

(19)システムプロセッサと、

予め決められた最小番号から最大番号までの複数のブロックを記憶することができ、そのうち番号が第1の値以上のブロックにBIOSが記憶されている記憶装置と、

前記システムプロセッサと前記記憶装置の間に接続され、前記システムプロセッサからのブロック形式の入出力要求を前記記憶装置の物理的特性に合った形式の要求に変換する制御手段と、

リセット信号の発生を開始する第1論理手段と、  
前記BIOSへのアクセスを禁止する第2の信

号を発生する第2論理手段と、

前記リセット信号に応答して前記BIOSへのアクセスを許可し、前記第2の信号に応答して前記第1の値のブロックのところに境界を設定することにより前記BIOSへのアクセスを禁止する保護手段と、

を具備するパーソナルコンピュータシステム。

(20)前記記憶装置はシリンドーヘッドーセクタ形式の要求を受けける固定ディスクであり、前記制御手段は前記ブロック形式を前記シリンドーヘッドーセクタ形式に変換する、請求項19に記載のシステム。

(21)前記第1論理手段は電源投入またはキーボード入力に応答して前記リセット信号の発生を開始する、請求項19に記載のシステム。

(22)システムプロセッサ、システムを初期設定するための手段を含む第1BIOS部分を記憶する読取専用メモリ、ランダムアクセスメモリおよび直接アクセス記憶装置を含むパーソナルコンピュータシステムにおいてBIOSを保護するための方

法であつて、

実行可能コードセグメントを含むマスタブートレコードと、前記パーソナルコンピュータシステムの通常動作中は前記ランダムアクセスメモリに常駐している第2BIOS部分を前記直接アクセス記憶装置の保護可能な区画に記憶し、

システムを初期設定して、前記直接アクセス記憶装置に送られるリセット信号の発生を開始し、

前記リセット信号に応答して前記保護可能な区画の保護を解除することにより、前記システムプロセッサが前記マスタブートレコードおよび前記第2BIOS部分をアクセスできるようにし、

前記マスタブートレコードを前記ランダムアクセスメモリにロードし、

制御を前記実行可能コードセグメントに移して、前記第2BIOS部分を前記ランダムアクセスメモリにロードし、

制御を前記ランダムアクセスメモリにある第2BIOS部分に移して、前記保護可能な区画へのアクセスを禁止する、

ことを特徴とするBIOS保護方法。

(23) 前記第1 BIOS部分に含まれるデータと、前記マスタートレコードに含まれる対応するデータとを比較することにより、前記マスタートレコードがシステムに適合しているかどうかを確かめる、請求項22に記載の方法。

(24) 前記読取専用メモリに含まれるデータと、前記マスタートレコードに含まれる対応するデータとを比較することにより、前記マスタートレコードが前記システムプロセッサに適合しているかどうかを確かめる、請求項22に記載の方法。

### 3. 発明の詳細な説明

#### A. 産業上の利用分野

本発明はパーソナルコンピュータシステム（以下、PCSという）に関し、特にその大容量記憶装置に記憶されているBIOSを保護するための方法及び装置に関する。

#### B. 従来の技術

PCSは現代社会の様々な分野で広く使用されるようになってきた。PCSは一般にデスクトップ

型、フロアスタンド型、ポータブル型等に分けられるが、いずれもシステムプロセッサ、ディスプレイモニタ、キーボード、1以上のディスケットドライブ、固定ディスク記憶装置、（オプションの）プリンタ等を備えている。このようなシステムの1つの特徴は、マザーボードあるいはシステムプレーナを用いてそれらの構成要素を電気的に接続していることである。PCSは主としてシングルユーザー向けに設計されており、個人あるいは中小企業のユーザーが手軽に買えるように、その価格も低く設定されている。

PCSはバスアーキテクチャによって2つのファミリにわけることができる。第1のファミリはファミリ1モデルと呼ばれ、IBM PC ATに代表されるようなPCSを含む。第2のファミリ、すなわちファミリ2モデルは、IBM PS/2モデル50ないし80のように、マイクロチャネルバスアーキテクチャを使用する。

ファミリ1モデルの初期のPCSでは、ソフトウェアの互換性が最も重要であると考えられてい

た。そのため、システム常驻コード（マイクロコード）という分離手段がハードウェアとソフトウェアの間に設定された。このコードは、ユーザーのアプリケーションプログラム／オペレーティングシステムと装置（デバイス）の間の動作インタフェースを与え、ハードウェアデバイスの特性についてユーザーが注意を払わなくてもよいようにしていた。最終的には、アプリケーションプログラムをハードウェアの特性から独立させつつ、新しいデバイスをシステムに付加できるようにするため、コードはBIOSへと発展していった。BIOSは、特定のデバイスハードウェア特性に対するデバイスドライバの依存性をなくすと共に、デバイスへの中間インタフェースとしての働きをデバイスドライバに持たせたため、その重要性が直ちに認められた。BIOSはシステムと一体化されていて、システムプロセッサに対するデータの入出力を制御するものであるから、読取専用メモリ（ROM）の形でシステムプレーナに組み込まれていた。例えば、初期のIBMパーソナルコン

ピュータのBIOSは、プレーナボード上のROMの8Kの領域を占めていた。

#### C. 発明が解決しようとする課題

PCSの新しいモデルが開発される時は、新しいハードウェア及びI/O装置を導入するためにBIOSを拡張しなければならなかった。当然のように、BIOSを記憶するためのメモリ容量は大きくなっていった。例えば、IBMパーソナルコンピュータATの場合、BIOSは32KバイトのROM容量を必要とした。

最近新しい技術の発達により、ファミリ2モデルのPCSは益々精巧になり、一般の消費者も使い始めている。技術の進歩は急速であり、新しいI/O装置がPCSに使用されたので、PCSの開発サイクルでBIOSの変更が問題になってきた。

例えば、マイクロチャネルアーキテクチャを採用したIBMパーソナルシステム/2の場合、新しい拡張BIOS（ABIOS）が開発された。しかしソフトウェアの互換性を維持するため、フ

ファミリ-1モデルからのBIOSをファミリ-2モデルに含ませる必要があった。ファミリ-1のBIOSは、互換BIOS(CBIOS)と呼ばれるようになった。しかしIBMパーソナルコンピュータATに関連して説明したように、プレーナボード上には32KバイトのROMしかなかった。システムは96KバイトのROMまで拡張できたが、システムの制約上、これがBIOSに対して使用できる最大の容量であった。幸いなことに、ABIOSを追加しても、ABIOS及びCBIOSの両方を96KのROMに詰め込むことができたが、残りの容量はごく僅かであった。従って、将来新しいI/O装置が付加されるときは、CBIOS及びABIOSはROMスペースを超過してしまい、新しいI/O技術を容易にはCBIOS及びABIOSに組み込めなくなる。

上述のような問題に加えて、ファミリ-2BIOSの変更を開発サイクルの出来るだけ遅い段階で行いたいという要求があったため、BIOSの部分をROMから外すことが必要になった。外さ

れた部分は、固定ディスクのような大容量記憶装置に記憶された。ディスクは読み書きが可能のため、実際のBIOSコードをディスク上で変更することが出来るようになった。ディスクはBIOSコードを記憶する高速で効率の良い手段であるが、BIOSコードが改悪される可能性が高くなった。BIOSはオペレーティングシステムと一体化されているため、改悪されたBIOSは破壊的な結果を招き、多くの場合システムの完全な障害及びノーオペレーションを引き起こす。従って、固定ディスクに記憶されているBIOSコードが許可なく変更されないようにするための手段が強く要求されており、そのような手段を提供することが本発明の目的の1つである。

#### D. 課題を解決するための手段

本発明に従うPCSは、システムプロセッサ、ランダムアクセスメモリ、ROM、及び少なくとも1つの直接アクセス記憶装置を含んでいる。システムプロセッサと直接アクセス記憶装置の間に接続される直接アクセス記憶装置コントローラは、

その記憶装置の一領域を保護するための手段を含む。保護される領域は、マスタブートレコード及びBIOSイメージを記憶する。保護手段はリセット信号に応答して保護領域へのアクセスを許可し、マスタブートレコードをランダムアクセスメモリへロードさせる。マスタブートレコードは動作時にBIOSイメージをランダムアクセスメモリにロードする。かくしてランダムアクセスメモリにあるBIOSが実行され、保護手段を活動化する信号を発生する。これにより、マスタブートレコード及びBIOSイメージを含むディスク上の領域へのアクセスが禁止される。次いでBIOSは、システムの動作を開始させるためオペレーティングシステムをブートアップする。

ROMはBIOSの第1部分を含んでいる。この第1部分はシステムプロセッサ及び直接アクセス記憶装置を初期設定し、保護手段をリセットして、直接アクセス記憶装置の保護領域にあるマスタブートレコードをランダムアクセスメモリに読みこませる。マスタブートレコードはデータセグ

メント及び実行可能コードセグメントを含む。データセグメントはシステムハードウェアと、マスタブートレコードがサポートするシステム構成を表わすデータを含む。第1BIOS部分は、マスタブートレコードのデータセグメントからのデータと、システムプロセッサ、システムプレーナ及びプレーナI/O構成を表わす第1BIOS部分内のデータとが一致していることを検証することにより、マスタブートレコードがシステムハードウェアに合っていることを確かめる。

マスタブートレコードがシステムハードウェアに合っていれば、第1BIOS部分は、システムプロセッサにマスタブートレコードの実行可能コードセグメントを実行させる。実行可能コードセグメントは、システム構成が変わっていないことを確かめ、BIOSの残りの部分を直接アクセス記憶装置からランダムアクセスメモリにロードする。次いで実行可能コードセグメントは、残余BIOS部分の真正さを検査し、システムプロセッサにランダムアクセスメモリにあるBIOSの実行を

開始させる。ランダムアクセスメモリ中のBIOSは、残りのBIOSを含むディスク区画を保護するための信号を発生し、オペレーティングシステムをブートアップして、PCSの動作を開始させる。残りのBIOSを含む区画を保護するのは、ディスク上のBIOSコードへのアクセスを禁止して、その完全性を守るためである。

#### E. 実施例

図面、特に第2図において、複数のI/Oスロット18を介してシステムボードないしプレーナボード24に接続した複数のDASD（直接アクセス記憶装置）12～16を有するPCS（パーソナルコンピュータシステム）10の一部を切除したものが示されている。電源22は周知のようにシステム10に電力を供給する。プレーナボード24は、入力、プロセスおよび出力情報に対し計算機命令による制御のもとで動作するシステムプロセッサを含んでいる。

使用に当り、PCS10は主として少数のユーザーのグループまたはシングルユーザーに個々に

計算パワーを与えるように設計されそして個人または小企業用に安価なものとされている。動作を述べると、このシステムプロセッサはIBMのOS/2またはPC-DOSのようなオペレーティングシステムのもとで動作する。この形式のオペレーティングシステムはDASD12～16とオペレーティングシステムの間にはBIOSインターフェースを含む。機能により複数のモジュールに分割されたBIOSの一部はプレーナ24上のROMに記憶され、そして以降ではROM-BIOSと呼ぶ。BIOSはハードウェアとオペレーティングシステムのソフトウェアの間にインターフェースを与え、プログラマまたはユーザが特定の装置について深いオペレーティング上の知識を必要とせずにそのマシンをプログラムしうるようにする。例えば、BIOSディスクットモジュールはプログラマがディスクット駆動ハードウェアの深い知識がなくともディスクット駆動機構をプログラムしうるようにする。このように異なる製造業者により設計され製造された多数のディスクット

駆動機構がこのシステムで使用出来る。これはシステム10のコストを低減するばかりでなくユーザが多数のディスクット駆動機構からそれを選ぶことが出来るようにする。

上記の構造を本発明に関連づける前に、PCS10の一般的な動作を要約する。第1図は、このPCS10のブロック図である。第1図はプレーナ24の構成要素と、I/Oスロット18およびこのシステムの他のハードウェアへのプレーナ24の接続を示している。プレーナ24の上にはシステムプロセッサ26が配置され、このプロセッサは局所母線28により、メモリコントローラ30に接続するマイクロプロセッサからなり、コントローラ30はランダムアクセスメモリ(RAM)32に接続する。このマイクロプロセッサは適当なものでよいが、インテル社の80386がその一例である。

本発明は以降において第1図について説明するが、本発明の装置および方法は他のプレーナボードハードウェア構成にも使用されるものである。

例えば、システムプロセッサはインテル80286または80486でもよい。

このプロセッサによりプレーナ識別番号(プレーナID)がアクセス可能である。プレーナIDはそのプレーナに固有のものであり使用されているプレーナの型式を識別する。例えば、プレーナIDはシステムプロセッサ26のI/Oポートを介して、またはスイッチを用いて読取るべくハードワイヤドしうる。また、ディスクコントローラへのプレーナ論理回路を用いて、システムプロセッサ26の別のI/Oポートによりリセット信号を発生させることができる。例えば、リセット信号発生のためにそのI/Oポートをアドレスしてプレーナ論理を活動化するソフトウェアにより、リセット信号の発生を開始することができる。

局所母線28は更に母船コントローラ34を通じてプレーナ24上の読取専用メモリ(ROM)36に接続する。

付加的な不揮発性メモリ(NVRAM)58が直列/並列ポートインターフェース40および母



線コントローラ34を介してマイクロプロセッサ26に接続する。この不揮発性メモリ58はシステムの電源が切れたときにも情報を維持するために電池でバックアップされたCMOSでよい。ROMは通常プレーナ上にあるから、ROMに記憶されたモデル値およびサブモデル値はシステムプロセッサおよびシステムプレーナI/O構成を夫々識別するために用いられる。このようにこれらの値はプロセッサとプレーナI/O構成を物理的に識別する。NVRAM58はシステム構成データを記憶するために用いられる。すなわち、このNVRAMはシステムの現在の構成を記述する値を含む。例えばNVRAMは固定ディスクまたはディスクットの容量、ディスプレイの型式、メモリ容量、時刻、日付等を記述する情報を含む。更に、ROMに記憶されたモデル値およびサブモデル値は構成設定のような特殊な構成プログラムが実行されるときにNVRAMにコピーされる。構成設定プログラムの目的はシステムの構成を特徴づける値をNVRAMに記憶させることである。この

ように適正に構成されたシステムについてのNVRAM内のモデル値およびサブモデル値はROMに記憶されたモデル値およびサブモデル値に夫々等しい。これらの値が等しくない場合には、それはそのシステムの構成が変更されていることを示す。第6D図により、BIOSのローディングとノ組合せにおいてこの特徴を詳述する。

第1図の説明を続けると、母線コントローラ34はI/Oプレーナ母線43によりI/Oスロット18、直列/並列インターフェース40および周辺装置コントローラ42に接続する。周辺装置コントローラ42は更にキーボード44、マウス46、診断パネル47、およびディスクットコントローラ64に接続する。NVRAM58の他に、直列/並列インターフェース40はプリンタ、ハードコピー装置等に情報を入/出力するための直列ポート48と並列ポート50に接続する。周知のように局所母線28はキャッシュコントローラ52、キャッシュメモリ68、コプロセッサ54およびDMAコントローラ56にも接続しうる。

およびDMAコントローラ56にも接続しうる。

システムプロセッサ26はその内部動作並びにPCSI0の他のエレメントとのインターフェースを制御する。例えば図示のシステムプロセッサ26は、固定ディスク駆動機構62のようなDASDに接続した小型コンピュータシステムインターフェース(SCSI)I/Oカード60に接続する。SCSIディスク駆動機構以外のものを本発明により固定ディスクとして使用出来る。固定ディスク62に加えて、システムプロセッサ26はディスクット駆動機構66を制御するディスクットコントローラ64にもインターフェースしうる。ここで「ハードファイル」は固定ディスク駆動機構62であり、「フロッピー」はディスクット駆動機構66を意味するものである。

本発明の以前にはROM36はハードウェア周辺装置に対しオペレーティングシステムをインターフェースするBIOSコードのすべてを含むことが出来た。本発明によればROM36はBIOSの一部のみを記憶するようにされる。この部分は

システムプロセッサ26により実行されるとき、固定ディスク62またはディスクット66からBIOSの第2のすなわち残りの部分、以下BIOSイメージと呼ぶ部分を入力する。このBIOSイメージは第1BIOS部分にとつて代わり、そしてこのシステムの一体化部分としてRAM32のような主メモリ内に常駐する。ROM36に記憶されたBIOSの第1部分(ROM-BIOS)は第3~4図で一般的に、そして第6A~6D図で詳細に説明する。BIOSの第2部分(BIOSイメージ)は第5図で説明し、BIOSイメージのローディングは第7図で説明する。DASDからのBIOSイメージのローディングによる他の利点はシステムプロセッサのRAM32にBIOSを直接にロード出来るということである。ROMのアクセスはRAMのそれより著しく高速であるから、コンピュータシステムの処理速度に大きな改善が得られる。

ROM36内のBIOSのオペレーションそして固定ディスクまたはディスクットからのBIO

Sイメージのローディングオペレーションを次に述べる。一般に、ROM-BIOSのような第1プログラムがシステムのプリチェックを行い、そしてRAMにBIOSマスタブートレコードをロードする。マスタブートレコードは検証情報を有するデータセグメントと実行可能なコードを有しローディング手段として働くコードセグメントを含む。この実行可能なコードはハードウェアの適合性とシステム構成の妥当性を判断するためにデータ情報を使用する。ハードウェアの適合性と適正なシステム構成についてのテスト後に、この実行可能なコードがRAMにBIOSイメージをロードし、主メモリ常驻プログラムを生成する。BIOSイメージはROM-BIOSに続き、そしてマシンのオペレーションを開始するためにオペレーティングシステムをロードする。説明の便宜上、マスタブートレコードの実行可能コードセグメントをMBRコード、データセグメントをMBRデータと呼ぶ。

第3図はROM-BIOSを構成している異つ

たコードモジュールを示すメモリマップを示す。ROM-BIOSは電源投入自己検査(POST)ステージIモジュール70、初期BIOSロード(IBL)ルーチンモジュール72、ディスケットモジュール74、ハードファイルモジュール76、ビデオモジュール78、診断パネルモジュール80およびハードウェア適合性データ82を含む。要するに、POSTステージI70はシステムの事前初期設定とテストを行う。IBLルーチン72はBIOSイメージがディスクまたはディスケットからロードされるべきかを決定し、適合性をチェックし、そしてマスタブートレコードをロードする。ディスケットモジュール74はディスケット駆動機構についての入出力機能を与える。ハードファイルモジュール76は固定ディスク等へのI/Oを制御する。ビデオモジュール78はビデオディスプレイに接続するビデオI/Oコントローラへの出力機能を制御する。診断パネルモジュール80はシステム用の診断ディスプレイ装置に対する制御を与える。ハードウェア適合性デー

タ82は第5図で述べるシステムモデル値およびサブモデル値のような値を含む。

第4図は固定ディスクまたはディスケットからシステムにBIOSイメージをロードするためのプロセス全体を示す。システムに電源が投入されると、システムプロセッサがステップ100においてPOSTステージIの入口点へと導かれる。POSTステージIはシステムを初期設定し、選択されたDASDからのBIOSイメージのロードに必要なシステム機能のみをステップ102でテストする。特にPOSTステージIは、必要であればプロセッサ/プレーナ機能、診断パネル、メモリサブシステム、割込みコントローラ、タイマー、DMAサブシステム、固定ディスクBIOSルーチン(ハードファイルモジュール76)およびディスケットBIOSルーチン(ディスケットモジュール74)を初期設定する。

POSTステージIがシステムを事前初期設定した後に、POSTステージIは初期BIOSロードモジュール72に含まれる初期BIOSロード

(IBL)ルーチンにシステムプロセッサを導く。このIBLルーチンはまずBIOSイメージが固定ディスクに記憶されているかあるいはディスケットからロードされるかを決定し、次にステップ104において選択された媒体(ディスクかディスケットか)からRAMにマスタブートレコードをロードする。マスタブートレコードはMBRデータとMBRコードを含む。MBRデータは検査用であり、MBRコードはBIOSイメージのロードのために実行される。IBLルーチンの詳細は第6A~6D図に示してある。

第4図において、IBLルーチンがマスタブートレコード(MBR)をRAMにロードした後に、ステップ106において、システムプロセッサがMBRコードのスタートアドレスを用いて実行を開始する。MBRコードはBIOSイメージの確認とシステム構成の検査のための一連の妥当性テストを行う。このMBRコードの詳細は第7図に示してある。

これら妥当性テストにもとづき、MBRコード

はBIOSイメージをRAMにロードし、そしてステップ108において制御を主メモリに新しくロードされたBIOSイメージへと移す。特に、BIOSイメージは前にROM-BIOSが占めていたアドレススペースにロードされる。すなわちROM-BIOSが0000HからFFFFFFHまでの間でアドレスされるとすると、BIOSイメージはこのRAMアドレススペースにロードされ、かくしてROM-BIOSに代えられる。次に制御は新しくロードされたBIOSイメージに含まれるPOSTステージIIに移され、かくしてROM-BIOSは放棄される。この後、RAMにあるPOSTステージIIはオペレーティングシステムをロードするためのステップ110~114を実行する。オペレーティングシステムに制御を移す前にステージII POSTは、BIOSイメージを保持するディスク区画へのアクセスを禁止するための保護手段をセットする。保護の詳細は第8~10図のところで説明する。ウオームスタート中にプロセッサはステップ100~106

がロードされるとき計算された検査合計値と比較される検査合計値132によりテストされる。データセグメントは更に少くとも1個の適合プレーナID値134、適合モデル値およびサブモデル値136を含む。マスタブートレコードのプレーナID値はマスタブートレコードが有効なプレーナを規定する。同様にマスタブートレコードのモデル値およびサブモデル値はマスタブートレコードが有効であるプロセッサとプレーナI/O構成を夫々規定する。ブートレコード識別子と検査合計は有効なマスタブートレコードを識別し、そしてブートレコードのプレーナID、モデル値およびサブモデル値の比較は、システムに適合するブートレコードの識別とシステム構成の有効性の決定に用いられる。他の値であるブートレコードパターン124はROM-BIOSの妥当性の決定に用いられる。ブートレコードパターン124はROMに記憶された対応するパターン値と比較される。それらが一致することは、有効なROM-BIOSが選ばれた媒体からのBIOSイメージのロー

をバイパスしてステップ108に導かれる。

ここでマスタブートレコードの形式について説明しておく。第5図はマスタブートレコードの形式を示す。ブートレコードは実行可能コードセグメント120とデータセグメント122~138を含む。MBRコード120はROM-BIOSの識別検査、IBLブートレコードがシステムと両立しうるかどうかのチェック、システム構成の検査および選ばれたDASD（ディスクまたはディスクセット）からのBIOSイメージのロードを実行するためのDASDに依存するコードを含む。データセグメント122~138は媒体の定義、マスタブートレコードの識別と検査、BIOSイメージの位置決めおよびBIOSイメージのロードに用いられる情報を含む。

マスタブートレコードはブートレコード識別子122により識別される。ブートレコード識別子122はレコードのはじめの3バイト内の文字列「ABC」のような固有のビットパターンである。マスタブートレコードの保全性は、ブートレコー

ドを開始したことを示す。

以下にマスタブートレコードの夫々の値とそれらの機能を詳述する。

MBR識別子(122)： IBLブートレコードのはじめの3バイトは「ABC」のような文字からなりうる。これはブートレコードの識別に用いられる。

MBRコードセグメント(120)： このコードは対応するプレーナIDとモデル/サブモデル値の比較により、プレーナおよびプロセッサに対するブートレコードの適合性を検査する。これらの値の一致は選ばれた媒体からシステムRAMへのBIOSイメージのロードを生じさせる。システムイメージ（メモリにロードされたBIOSイメージ）検査合計が有効であり媒体ロードエラーが生じないならば、MBRコードは制御をシステムイメージのPOSTステージIIルーチンに移す。

MBRパターン(124)： IBLブートレコードデータセグメントのはじめのフィールドは文字列「ROM-BIOS 1989」のようなパター

ンを含む。この文字列は、ブートパターン値とROM内に記憶された対応する値（ROM-パターン）を比較することによりROM-BIOSの妥当性チェックに用いられる。

MBR版デート（126）： マスタブートレコードは更新のための版デートを含む。

システム区画ポインタ（128）： データセグメントはステージII POSTで用いるための媒体システム区画領域の開始点を示す媒体ポインタを含む。IBLディスクではこのポインタはトラック-ヘッド-セクタ形式であり、ディスクでは相対ブロックアドレス（RBA）形式である。

システム区画タイプ（130）： システム区画タイプは媒体システム区画の構造を示す。3つのシステム区画構造のタイプ、すなわち、完全、最小および不在、がある。完全システム区画はBIOSイメージおよびマスタブートレコードに加えてセクタアップユーティリティおよび診断を含む。最小システム区画はBIOSイメージとマスタブートレコードのみを含む。システムがIBLイ

メージを有するハードファイルへのアクセスを有さないことがあり、この場合にはシステム区画タイプは不在となる。この例ではIBLはディスクから生じることになる。これら三種のシステム区画タイプにより、媒体上でシステム区画が占めるスペースを変えることができる。

検査合計値（132）： データセグメントの検査合計値は、MBRコードのレコード長値（15kバイト）についての有効検査合計を発生するために初期設定される。

MBRブレーナID値（134）： データセグメントは適合ブレーナIDを規定するワード列のような値を含む。各ワードは16ビットのブレーナIDからなり、この列はワード値0で終る。システムのブレーナIDがマスタブートレコード内のブレーナID値と一致すると、IBL媒体イメージはシステムブレーナに適合しうる。システムのブレーナIDが列内のいずれのワードとも一致しないならば、IBL媒体イメージはシステムブレーナに適合しない。

MBRモデル値およびサブモデル値（136）： このデータセグメントは適合プロセッサを規定するワード列のような値を含む。各ワードはモデル値およびサブモデル値からなり、このワード列はワード値0で終る。システムのモデル値およびサブモデル値（ROMに記憶されたもの）がこのワード列内の1つのワードと一致するならばIBL媒体イメージはそのシステムプロセッサに適合する。ROMモデル値とROMサブモデル値がこのワード列中のいずれのワードとも一致しないならば、IBL媒体イメージはシステムプロセッサに適合しない。

MBRマップ長（138）： IBLマップ長は媒体イメージブロックの数に初期設定される。言い換えると、BIOSイメージが4個のブロックに分割されるならばマップ長は4となり、4個のブロックポインタ/長さフィールドを示す。媒体イメージは1個の連続した128kブロックであるから一般にこの長さは1にセットされる。

MBR媒体セクタサイズ（138）： このワード

値は媒体セクタサイズ（セクタのバイト数）に初期設定される。

媒体イメージブロックポインタ（138）： この媒体イメージブロックポインタは媒体上でシステムイメージブロックを見つけ出す。通常は、媒体イメージが1個の連続したブロックとして記憶されるため、1個のポインタしか存在しない。IBLディスクではポインタはトラック-ヘッド-セクタ形式となっており、ディスクでは相対ブロックアドレス形式となっている。

媒体イメージブロック長（138）： 媒体イメージブロック長は対応するイメージブロックポインタが示すブロックのサイズ（セクタ数）を示す。BASIC用のスペースを含む128kの連続する媒体イメージの場合には、このフィールドは256にセットされて、BIOSイメージブロックが媒体イメージブロックポインタ位置からはじまり256個のセクタ（512バイト/セクタ）からなることを示す。

第6A～6D図はIBLルーチンの詳細なフロー

チャートである。通常状態ではIBLルーチンはシステムの固定ディスクからRAMの特定のアドレスにマスタブートレコードをロードし、そしてシステムプロセッサをこのマスタブートレコードのコードセグメントの実行開始へと導く。またIBLルーチンは、マスタブートレコードがディスクセットからロードされうるディスクセットデフォルトモードを含む。しかしながら、IBLルーチンはシステムが固定ディスク上にIBL媒体を含みそして有効パスワードがNVRAMにあるならば、ディスクセットデフォルトモードの実行を許可しない。ユーザーはNVRAMにそのパスワードをセットしうる。ディスクセットデフォルトモードを禁止する目的は、ディスクセットからの許可されないBIOSイメージのロードを防止することである。言い換えると、ディスクセットデフォルトモードはシステムの固定ディスクが動作不能であり、ユーザーがディスクセットからのロードを望むことを示す（パスワードをセットしない）ときにのみ用いられる。IBLルーチンがいずれの媒体からのマスタ

ブートレコードのローディングもしえないならば、エラーメッセージが発生されてシステムが停止する。

第6A図において、通常、システムはシステム固定ディスクを含み、このディスクがIBLルーチンにより初期設定される（ステップ150）。固定ディスクがPCのドライブC用の構成とされているものとし、またドライブAがディスクセット駆動機構に割り当てられているとすると、IBLルーチンはドライブCがIBL媒体を含むかどうかの決定をステップ152で行う。このプロセスの詳細を第6B図に示す。IBLルーチンは固定ディスクの読取りを最後の3セクタのところから開始し、99セクタの間、あるいは有効マスタブートレコードが見い出されるまで媒体ポインタを減分しながら読取りを続ける。マスタブートレコードが見い出されたならば、ステップ156でシステムブレーナおよびプロセッサに対する適合性についてチェックする。適合性がなければステップ158でエラーが出される。ステップ152にお

いて固定ディスク（1次ハードファイル）の最後の99セクタにマスタブートレコードがなければ、ステップ154でエラーが出される。

ステップ156でマスタブートレコードがあれば一連の有効性チェックが行われ、マスタブートレコードがコンピュータシステムに適合しているかどうかを決定する。更に、このシステムの構成がチェックされる。このプロセスの詳細を第6D図に示す。ブートレコードがブレーナID、モデルおよびサブモデルに適合し、そして更にシステム構成が変更されていないならば、ステップ160でマスタブートレコードがロードされ、そしてそのコードセグメントが実行される。

ステップ154と158において、固定ディスクからマスタブートレコードをロードする際にエラーが生じあるいは固定ディスクが使用出来ない場合には、IBLルーチンはステップ162で有効パスワードがNVRAMに含まれているかどうかを決定する。このパスワードはBIOSイメージがディスクセットからロードされうるかどうかを

決定する。このパスワードは、セットアップユーティリティを動かすユーザーにより設定されているときにのみ存在する。パスワードがNVRAMに設定されていれば、ステップ164においてBIOSイメージはディスクセットからロードされないようにされる。これはユーザーに、固定ディスク上のBIOSイメージだけをシステムにロードするようにすることによりシステムのオペレーションの完全性を保証しうるようにする。このパスワードはNVRAMに記憶された文字列の形をとることが出来る。

ステップ162において、NVRAM内に有効パスワードがなく、BIOSイメージがディスクセットからロードしうる場合には、IBLルーチンがステップ166においてディスクセットサブシステムを初期設定する。このIBLルーチンはステップ168においてドライブAがディスクセット上にIBL媒体を含むかどうかを決定するドライブAがIBL媒体を含んでいなければステップ170において無効ディスクセットがドライブに挿入さ

れていることをユーザに知らせるためのエラーが発生する。ステップ168の詳細を第6C図に示す。

ステップ168において、ドライブAがIBL媒体についてチェックされた後に、ステップ160でマスタブートレコードがRAMにロードされ、そしてそのコードセグメントが実行される。ディスクセットについては、IBLルーチンは固定ディスクシステムに用いた妥当性チェックを含まない。例えばシステムに新しいプロセッサが加えられる場合には、新しいBIOSイメージがディスクセットに含まれることになる。新しいプロセッサは固定ディスクからのローディングについて妥当性エラーを生じさせるから、IBLルーチンはBIOSイメージをディスクセットからロードすることによりこれらテストをバイパスする能力を与える。

要点を繰返すと、マスタブートレコードはブートレコード値に対するシステムブレーナIDとプロセッサモデル/サブモデル値の整合により、システムとの適合性についてチェックされる。ディ

スクについてはこのチェックはまずIBLルーチン72で行われ、次にIBLブートレコードにおいて再び行われる。最初のチェック（IBLルーチンにおける）はブートレコードがシステムに適合することを確認するために行われ、2番目のチェック（ブートレコードにおける）は適合ROMが制御をブートレコードに移したことを確認するために行われる。ディスクセットブートレコードで行われるチェックは、IBLルーチンが適合性をすでにチェックしているから、適合ROMについては誤りはない。一方、最初の適合性チェックはディスクセットについては行われない。ブレーナ/プロセッサの適合性はディスクセットブートレコードの実行中にのみチェックされる。この方法により、基準ディスクセットからの新しいBIOSイメージのローディングにおける将来の変更が可能になる。

第6A図のIBLルーチンにおける妥当性テストを更に詳細に説明する。第6B図は第6A図における有効マスタブートレコードがドライブCに

あるかどうかの決定のためのステップ152の詳細なフローチャートである。このプロセスはステップ200でIBLルーチンがドライブCにアクセスしようとするドライブパラメータを得ることにより開始する。IBLロード位置がステップ202においてディスクから最後の3セクタ（これらセクタは通常マスタブートレコードを含む）にセットされる。ディスクからマスタブートレコードを読み取る試行回数を示すロードカウントがステップ204において1にセットされる。13ロード位置の3個のセクタがステップ206においてディスクから読取られる。ステップ208～210において、何らかのディスクドライブエラーが検出され、ディスク読取エラーが生じればそれが報告される。プロセスはエラー標識を伴ってリターンすることになる（ステップ212～214）。

ステップ208でドライブエラーが生じていなければ、そのディスクレコードがステップ216においてマスタブートレコード識別子について走

査される。文字「ABC」のようなこのブートレコード識別子はディスクレコードのはじめの3バイトと比較される。ディスクレコードが有効ブートレコード識別子（文字「ABC」）を有し、そしてメモリにロードされたディスクレコードから計算された検査合計がブートレコードの検査合計に等しければ、このディスクレコードはステップ218においてエラーを有さない有効ブートレコードとして示される。ステップ214においてこのプロセスは第6A図に戻る。

ステップ216においてブートレコード識別子または検査合計が無効であれば、ロードカウントがステップ220において1だけ増加される。このロードカウントはステップ222において99のような予定の定数と比較される。ブートレコードの読取を99回試み、不成功である場合にはステップ224、212および214においてエラーが示されリターンする。ブートレコードの読取りが99回より少いときはIBLロード位置がステップ226で1だけ減分され、そして3個の新し

いセクタがステップ206で新しいロード位置から読取られる。かくして最後の99個のセクタ(33コピーに等価)から有効IBLブートレコードがロードされえないときにはエラー条件がセットされて、制御がIBLルーチンに戻される。

マスタブートレコードをドライブAのディスクットからロードすることについての詳細を示す第6C図において、まずドライブAのアクセスのための、ディスクットドライブパラメーターがステップ230でとり出される。IBLロード位置がステップ232においてディスクットの最後の3個のセクタ(シリング、ヘッドおよびセクタ形式)セットされる。これら3個のセクタがステップ234で読取られる。ステップ236~238において、ディスクットドライブエラーが検出されるとエラーが示される。ステップ240~242において、エラー条件がセットされ、制御がIBLルーチンに戻される。

ステップ236においてドライブエラーが検出されないと、ディスクットレコードはステップ2

44においてブートレコード識別子をチェックされ、そして検査合計が計算される。ブートレコード識別子がなくあるいは検査合計が無効であれば、エラーが示されて制御がIBLルーチンにもどされる。有効ブートレコード識別子と有効検査合計が検出されると認識がセットされ(ステップ248)、そして制御がIBLルーチンに戻される。ディスクットロードの場合、IBLルーチンは固定ディスクロードにおけるようなセクタの探索を行わない。それ故ディスクットロードでは、IBL媒体はディスクットの特定の位置に記憶されねばならない。

最後に、第6D図はシステムブレーナとプロセッサの適合性および適正なシステム構成についてのIBLルーチンにおけるテスト方法を示す。ステップ260においてマスタブートレコードが、そのブレーナID値をシステムプロセッサにより読取られたシステムブレーナIDと比較することにより、システムブレーナとの適合性をチェックされる。システムブレーナIDがブートレコード

ブレーナID値と一致しないときは、このマスタブートレコードがこのブレーナに適合しないことを示す。ステップ262、264、266において、エラーが示され制御はIBLルーチンに戻る。

マスタブートレコードがブレーナと適合するのであれば、ステップ268においてプロセッサとの適合性についてマスタブートレコードがチェックされる。ブートレコードのモデル値およびサブモデル値がROMに記憶されたモデル値およびサブモデル値と夫々比較される。一致しないときは、新しいプロセッサが多分導入されており、このブートレコードがその新しいプロセッサに適合しないことを示す。ステップ270、264、266において、エラーが示されそして制御はIBLルーチンに戻る。マスタブートレコードがブレーナおよびプロセッサに適合するのであれば、ステップ272でNVRAMが信頼出来るかどうかについてのチェックを行う。NVRAMが信頼出来なければ、ステップ274、266において、エラーが示されそして制御がIBLルーチンに戻る。N

VRAMが信頼出来れば、ステップ276においてシステム構成がチェックされる。NVRAMに記憶されたモデル値およびサブモデル値がROMに記憶されたモデル値およびサブモデル値と一致しない場合は、システム構成が変更したことを示す。この最後の比較は構成エラーのみを示す。構成エラーが生じると、ユーザに対しエラーが知らされる。このエラーは、構成設定を最後にランさせた後にシステム構成が変更されたことをユーザに示すものである。ステップ278、264、266において、ユーザは変更された構成についての情報を受け、そして制御がIBLルーチンに戻される。このエラーは致命的なものではなく、構成設定(構成プログラム)を実行すべきことをユーザに知らせるものである。ステップ276でシステムモデル/サブモデル値が一致すると、ステップ274で適合性認識がセットされ、IBLルーチンに戻る。このように、マスタブートレコードとシステムの適合性は、システム構成が変化したかどうかの決定と並行してテストされる。

IBLルーチンがマスタブートレコードをRAMにロードした後に、制御がMBRコード開始アドレスに移される。第7図において、マスタブートレコードの実行可能コードセグメントがまずステップ300においてROMおよびブートレコードのパターンを検査する。マスタブートレコード内のパターンがROM内のパターンと一致しない場合にはステップ302においてエラーが発生され、ステップ305でシステムは停止する。ROMとブートレコードのパターンの一致についてのチェックは、ディスクまたはディスクセットからロードされたマスタブートレコードがブレーナボード上のROMに適合しているかどうかを確認する。ステップ300でROMのパターンがブートレコードのそれと一致すれば、MBRコードがステップ304でシステムブレーナID値、モデル値およびサブモデル値を対応するマスタブートレコード値と比較する。このプロセスは第6D図において述べた。これら値が一致しない場合には、マスタブートレコードがシステムブレーナおよびプロセ

ッサと適合しないこと、あるいはシステム構成が変わっていることを示し、ステップ306でエラーが発生される。そのときステップ305でシステムは停止する。

ステップ304においてシステムブレーナID値、モデル値およびサブモデル値が対応するマスタブートレコード値と一致すると、ステップ308でMBRコードが選ばれた媒体からシステムRAMにBIOSイメージをロードする。ステップ310でデータ読取において媒体ロードエラーが生じると、ステップ312においてエラーが発生されてシステムが停止する。ステップ310で媒体ロードエラーが発生しないと、ステップ314で検査合計がメモリ内のBIOSイメージについて計算される。この合計が無効であれば、ステップ318でエラーが発生されてシステムが停止する。ステップ316での検査合計が有効であれば、ステップ320でシステム区画ポインタが記憶され、そしてステップ322でシステムプロセッサがPOSTステージIIで移されてシステムのロー

ディングを開始する。

第8図は、ディスク駆動機構351とシステムプロセッサの間のデータ転送を制御する知能ディスクコントローラ350を示している。ディスクコントローラ350は第1図のアダプタカード60に組み込むことができるが、その場合は、ディスク駆動機構351は固定ディスク駆動機構62に対応することになる。本実施例のディスクコントローラ350は、本出願人が製造しているSCSIアダプタ(部品番号33F8740)である。ディスクコントローラ350に含まれるマイクロプロセッサ352は自身の内部クロックに従って動作し、その内部オペレーションだけでなく、ディスクサブシステムの他の構成要素およびシステムプロセッサとのインターフェースも制御する。マイクロプロセッサ352は命令母線354によりROM356に接続される。ROM356は、ディスク駆動機構とシステムプロセッサの間のデータ転送の処理および制御のためにディスクコントローラ350が実行する命令を記憶している。マ

イクロプロセッサ352には、データの記憶や検索のためのランダムアクセスメモリを接続することが出来る。ディスクコントローラ350とシステムプロセッサの間のデータ転送はデータ母線358および命令母線360により行われる。線362上のリセット信号は、電源投入時あるいはシステムリセット時にディスクコントローラ論理をリセットないし初期設定する。このリセット信号はブレーナボード論理で発生され、例えば本出願人が1987年5月に発行した"IBM PERSONAL SYSTEM/2 Seminar Proceedings"第5巻、第3号に記載されているようなマイクロチャネルアーキテクチャのチャネルリセット信号の形をとる。また、ブレーナ論理が接続されているシステムプロセッサのI/OポートにBIOSが特定のビット構成を出力することによってリセット信号を効果的に開始することが出来る。

周知のように、マイクロプロセッサ352は、ディスク駆動機構とシステムプロセッサの間で効



率の良いデータ転送を行うためのすべてのインターフェース信号及びタイミング信号を供給する。ここでは簡単のため、本発明の理解に必要な信号だけを取り上げる。また、ROM 356に記憶されているプログラムについても、本発明の理解に必要なものだけを第9図を参照しながら説明する。

第9図はディスクコントローラの読取り、書き込みおよび保護の機能を示すフローチャートで、ROM 356に記憶されているルーチンにより制御される。動作時には、ディスク命令がシステムプロセッサからディスクコントローラへ送られる。ディスクコントローラはステップ400でこの命令を受け取って解釈し、指定されたオペレーションを実行する。ディスクコントローラは、まずステップ402で、オペレーションがシステムプロセッサからディスクへのデータ書き込みかどうかを調べる。もし書き込みであれば、相対ブロックアドレス(RBA)形式でシステムプロセッサからデータを受け取る。

説明を続ける前にRBA形式について簡単に触

れておく。RBAは、ディスクのような大容量記憶装置のデータが所定サイズのブロックにおいてシーケンシャル番号でアドレス指定されるような方式のことである。例えばブロックサイズを1024バイトとすると、ディスク容量が10メガバイトの場合、システムプロセッサは約10000のブロックをアドレス指定出来る。本発明に従うPCS用のオペレーティングシステムにおいてRBAを用いると、大容量記憶装置を極めて高速に且つ効率良くアドレス指定することが出来る。

説明の便宜上、次のことを仮定する。第1に、ディスクは全部でN個のブロックをサポート出来る。第2に、システムプロセッサはブロックKを転送する。Kは0以上、N-1以下である。第3に、ディスクコントローラは最大アドレス可能ブロックMを設定することが出来る。これは、KがMより小さい場合にデータブロックへのアクセスを許可し、KがM以上の場合にデータブロックへのアクセスを拒否する。従って、MをNよりも小さい値に設定すると、MからN-1までのブロッ

クについてディスク上に保護領域が生成される。以下に述べるように、この特徴によりIBL媒体が保護される。

第9図に戻って、ステップ404でデータがRBA形式で受け取られる。次にディスクコントローラは、ステップ404において、受け取ったブロックの番号Kが最大ブロック値M(<N)より小さいかどうかを調べる。KがMより小さければ、ステップ408でディスクコントローラはRBA形式をシリンダーヘッドセクタ(CHS)形式のような大容量記憶装置特有の形式に変換する。例えば、ディスクコントローラは表引きを利用してRBAアドレスを一意的なCHS記憶位置に変換することが出来る。変換式を利用してRBAをCHSに変換する方法もある。例えば、1つのヘッド、64のシリンダーおよび96のセクタを有するディスクの場合、ヘッド=0、シリンダー=RBA/96の商、セクタ=RBA/96の剰余となる。RBA形式をCHS形式に変換した後、ステップ410でデータがディスクに変換されたCH

S記憶位置に書き込まれる。次にディスクコントローラはステップ412に進んで、システムプロセッサからの別の命令を待つ。

ステップ406に戻って、受け取ったRBAが設定された最大RBA値以上であれば、ステップ414でアクセスが拒否される。すなわち、KがM以上であれば、ブロックKはディスクには書き込まれない。したがって、もしIBL媒体がMからN-1までのブロックに記憶されていると、そのIBL媒体は書き込みから保護されることになる。

ステップ402に戻って、システムプロセッサからの命令が書き込みでなければステップ416で読取命令かどうかを調べる。読取命令であれば、システムプロセッサは要求されたデータについてのRBA形式を送っているから、ステップ418でそれを受け取る。後続のステップ420、422、424および426は、読取が行われることを除くと、前述のステップ406、408、410および414と同じである。従って、IBL媒体はコピーからも保護される。読取られたデータ

はステップ412でシステムプロセッサに転送される。

ステップ416に戻って、命令が読取でなければ、ステップ428で最大RBA設定命令かどうかを調べる。ディスクコントローラは、この命令により、ディスク上に保護領域ないし保護区画を生成することが出来る。この命令を受け取ると、ディスクコントローラはステップ430でMを0からNまでの間の値に設定する。ディスクコントローラが(リセット信号により)リセットされると、Mは最大数のブロックが使用可能になるように設定される。すなわち、ディスクコントローラがリセットされたときは、 $M=N$ である。基本的には、特定領域の保護は、ディスクコントローラがリセットされると解除され、その領域へのアクセスが可能になる。最大RBA設定命令が実行されてしまうと、リセットまたは別の最大RBA設定命令に依らない限り、保護領域へのアクセスは行えない。次いでディスクコントローラはステップ412へ進んで、別の命令を待つ。

第10図はIBL媒体の保護に関するフローチャートを示している。システムはステップ450の電源投入により初期設定され、BIOSがプレナボード論理で活動を開始して、ステップ452でディスクコントローラにリセット信号を送る。このリセット信号はIBL媒体の保護を解除し、システムプロセッサがそれをアクセス出来るようにする。IBL媒体はブロックMからブロックN-1までの領域で記憶されている。システムは第4〜7図のところで説明したようにして、ステップ454でIBL媒体をロードする。IBLローディングシーケンスの間にステップ456でPOSTステージIIが実行される。POSTステージIIのタスクの1つは、最大RBA設定命令を実行して、最大RBAをIBL媒体の最初のブロックの番号Mに設定することである(ステップ458)。Mは前に説明した区画のタイプ(完全、最小、不在)に依存する。これによりIBL媒体へのアクセスは禁止されるが、ディスクの他の領域へのアクセスは可能である。最後のステップ460でオペレー

ステップ428に戻って、命令が最大RBA設定でなければ、ステップ400で受け取った命令は、書き込み、読取および最大RBA設定以外の命令であり、ステップ432で実行される。そのような命令は本発明とは無関係であるから、ここでは触れないことにする。ディスクコントローラはステップ412へ進み、別の命令を待つ。

以上のことを考慮しながら、次にIBL媒体のローディングおよび保護のオペレーションについて説明する。一般に、IBL媒体を有するディスクコントローラは、コールドスタート(電源投入)またはウォームスタート(After-Cold-Start)でリセットされる。これにより、最大RBA(M)はNに設定され、IBL媒体へのアクセスが可能になる。こうしておかないと、システムはオペレーションを開始するためにIBL媒体をロードすることが出来なくなる。一旦IBL媒体がロードされて実行されると、ディスク上のIBL媒体へのアクセスを禁止するために、最大RBAがIBL媒体より下に設定される。

ティングシステムがブートアップされる。

ステップ462のウォームスタートの場合、プレナ論理は次のステップ464で、POSTステージIIによりディスクコントローラをリセットするよう指令される。これはIBL媒体の保護を解除する。この場合は、IBL媒体は既にRAMにあるので、それが再びロードされることはない。しかしIBL媒体の保護が解除されているため、POSTステージIIを実行することによつて再び保護を図る必要がある。そのためステップ458が実行され、ステップ460でシステムが再ブートされる。

要約すると、本発明は固定ディスクのような大容量記憶装置に記憶されているIBL媒体へのアクセスを保護するための方法および装置を提供するものである。IBL媒体の保護は、大容量記憶装置をブロック単位でアドレス指定し、システムが通常のオペレーション中にアクセス出来る最大ブロック番号を設定することで実現される。IBL媒体は、それよりも番号の大きいブロックの

こに連続的に記憶される。ディスクコントローラに送られるリセット信号は、最大ブロック値を解除し、システムがI B L媒体をアドレス指定できるようにする。リセット信号は、電源投入時またはウォームスタート時に発生される。

#### F. 発明の効果

本発明によれば、ディスク等の直接アクセス記憶装置に記憶されているB I O Sが許可なく変更されるのを効果的に防止することができる。

#### 4. 図面の簡単な説明

第1図は、本発明に従うP C Sの構成を示すブロック図。

第2図は、P C S本体の縦断斜視図。

第3図は、ROM-B I O Sのメモリマップを示す図。

第4図は、B I O Sイメージをロードするための全体的なプロセスを示すフローチャート。

第5図は、マスタブートレコードの形式を示す図。

第6 A図は、I B Lルーチンのオペレーション

を示すフローチャート。

第6 B図は、B I O Sイメージを固定ディスクからロードするためのステップを示すフローチャート。

第6 C図は、B I O Sイメージをディスクからロードするためのステップを示すフローチャート。

第6 D図は、マスタブートレコードとプレーナ/プロセッサの適合性をチェックする詳細なステップを示すフローチャート。

第7図は、実行可能コードセグメントのオペレーションを示すフローチャート。

第8図は、直接アクセス記憶装置コントローラを示すブロック図。

第9図は、I B L媒体を保護するためのディスクコントローラのオペレーションを示すフローチャート。

第10図は、B I O Sイメージを保護するための方法を示すフローチャート。

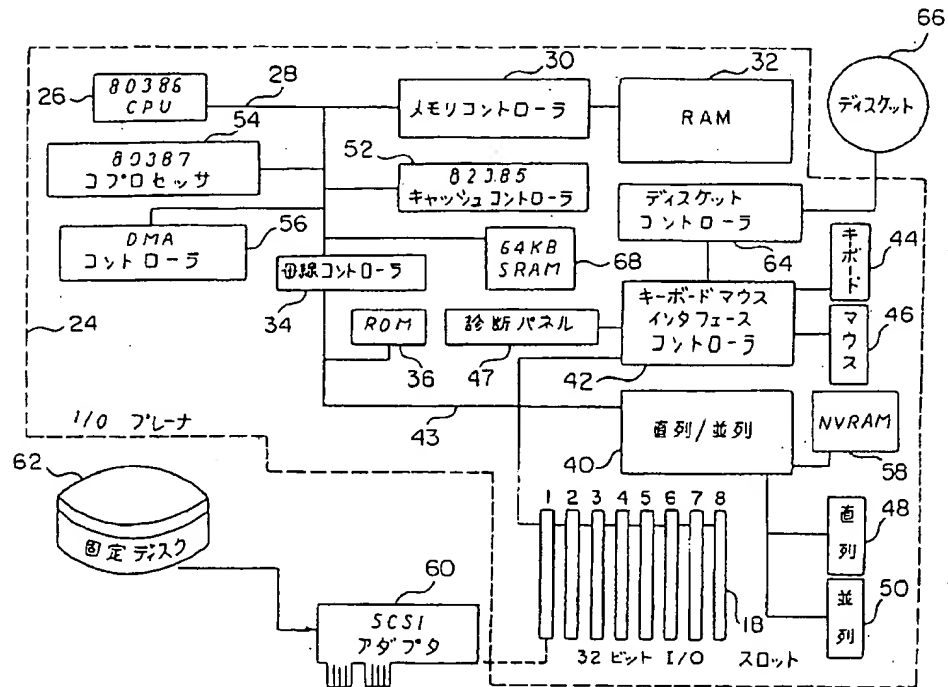


FIG. 1

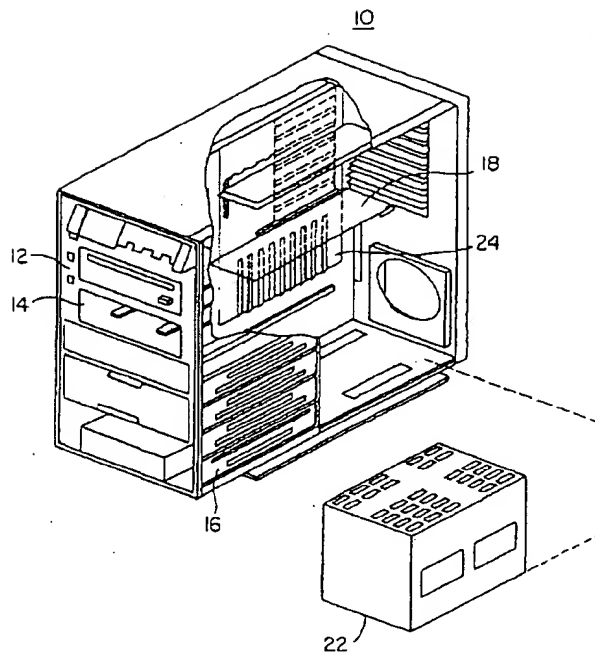


FIG. 2

POST ステージ 1	70
初期 BIOS ロードル・チン	72
ディスク	74
ハードファイル	76
ビデオ	78
診断パネル	80
ハードウェア 適合性データ	82

ROM - BIOS

FIG. 3

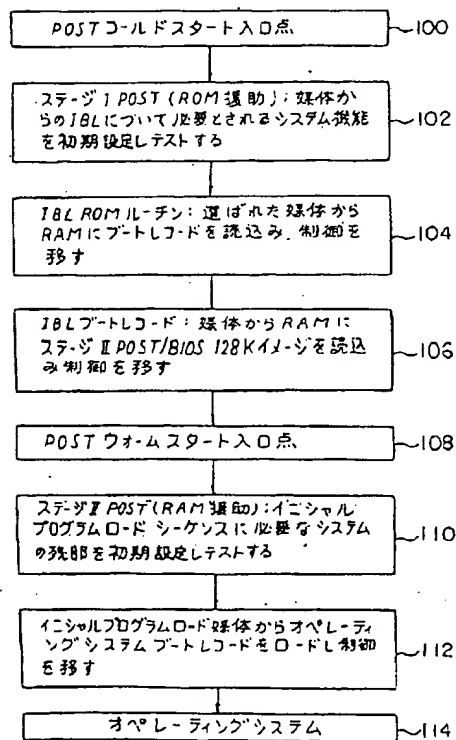


FIG. 4

MBR 識別子 "ABC"	122
マスタートレコード コードセグメント	120
...	
MBR パターン	124
MBR 版データ	126
システム区画	128
システム区画タイプ	130
MBR 検査合計	132
適合プレーナ ID	134
適合プロセッサモデル およびサブモデル バイト	136
MBR マップ長	138
MBR 媒体セクタサイズ	
第 1 ブロック ポインタ	
第 1 ブロック長	
第 2 ブロック ポインタ	
第 2 ブロック長	
...	
最終ブロック ポインタ	
最終ブロック長	

FIG. 5

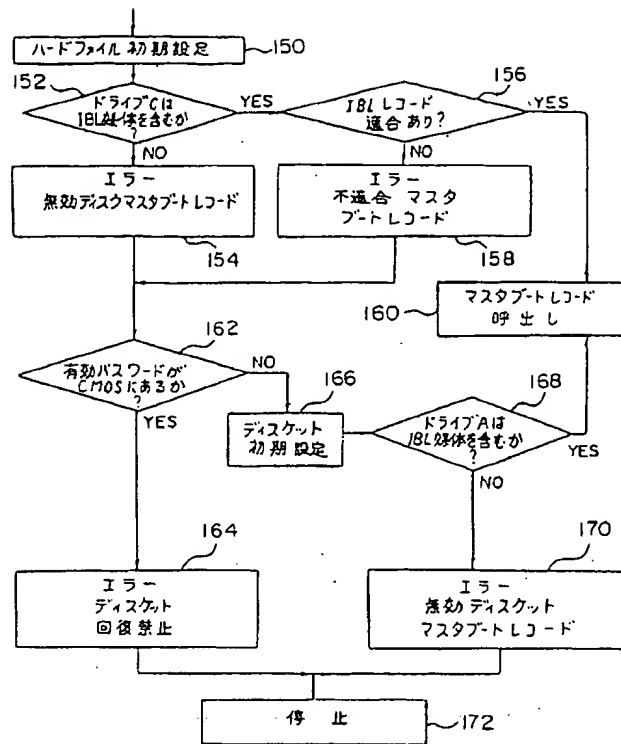


FIG. 6A

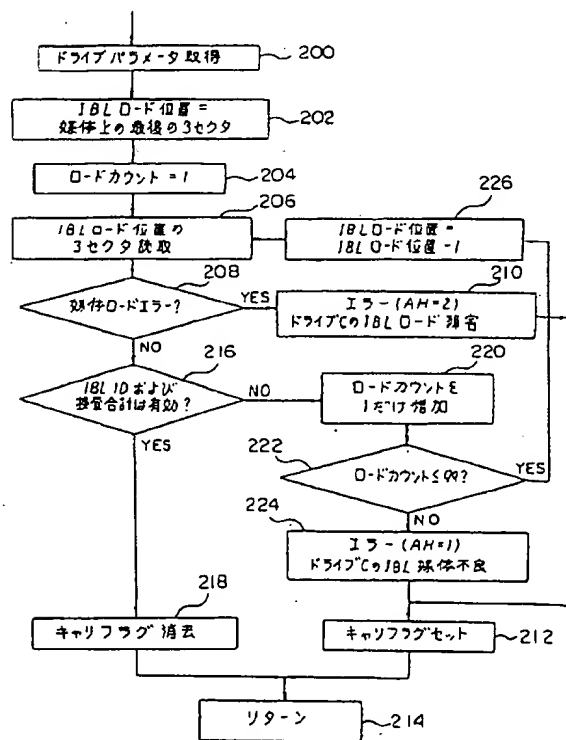


FIG. 6B

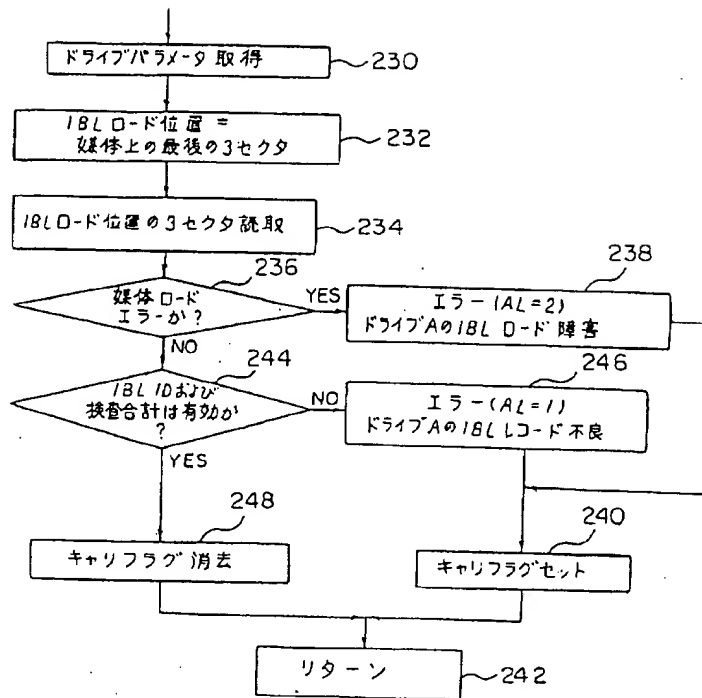


FIG. 6C

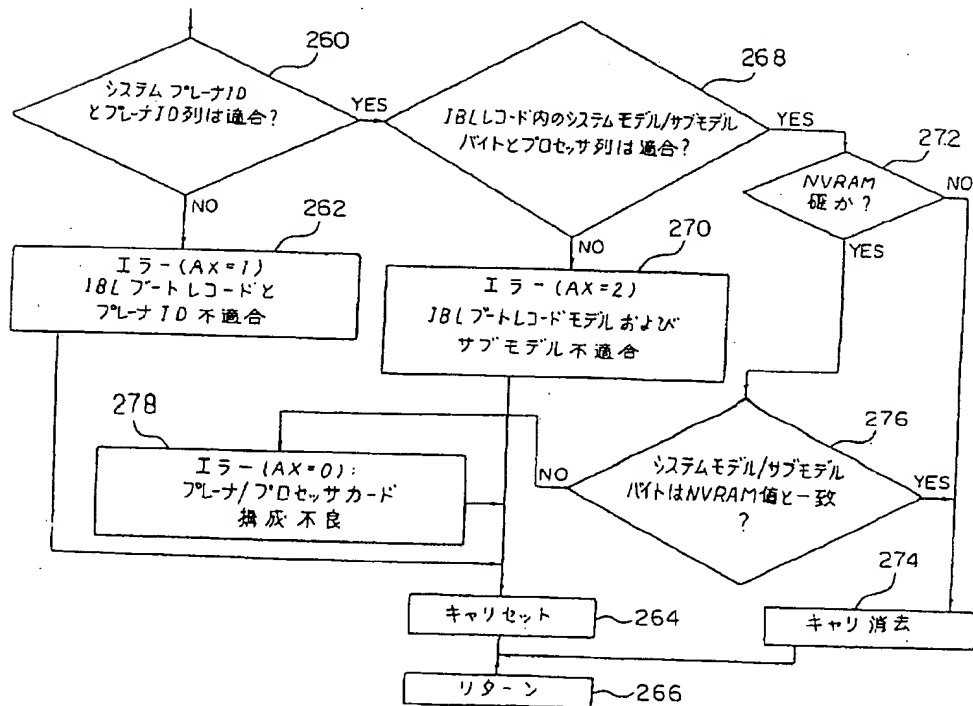


FIG. 6D

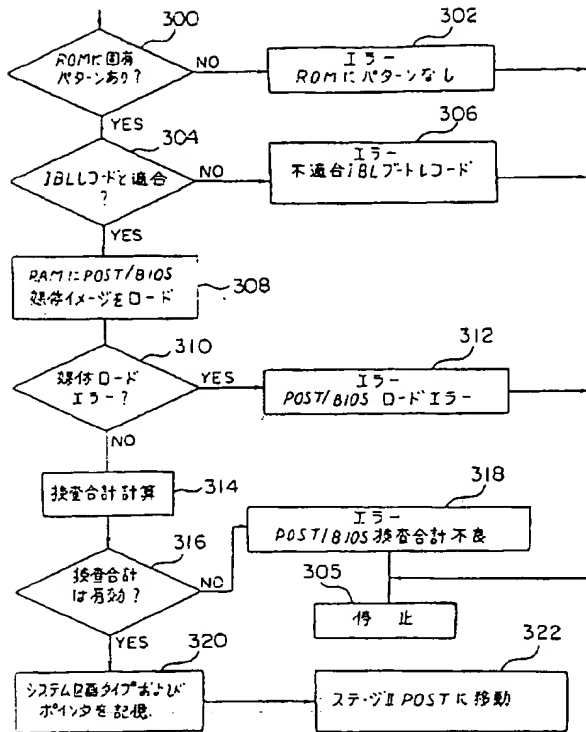


FIG. 7

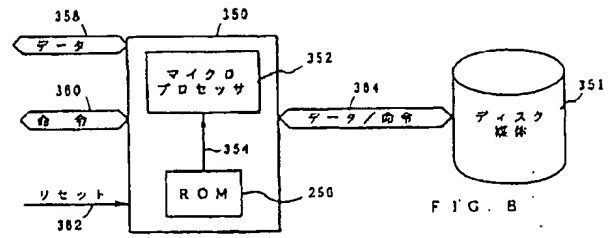


FIG. 8

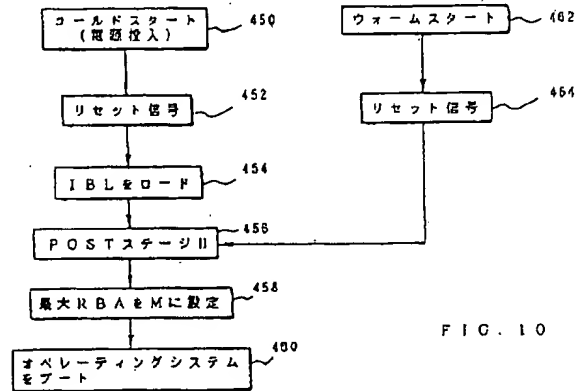


FIG. 10

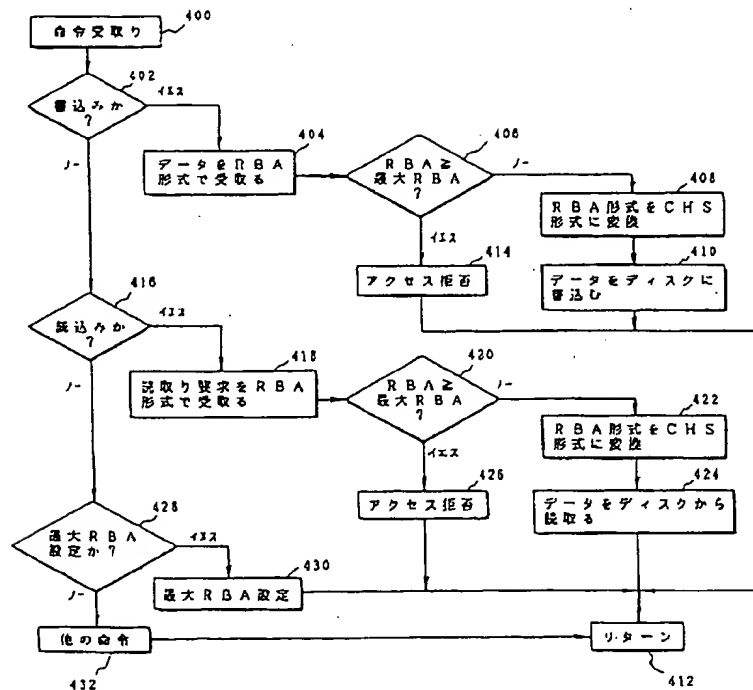


FIG. 9

第1頁の続き

⑨Int. Cl. <sup>3</sup>	識別記号	庁内整理番号
G 06 F 13/10	3 2 0 Z	7218-5B
⑦発明者	ドイル・スタンフィー ル・クロンク	アメリカ合衆国フロリダ州ボカ・ラトン、タウン・ハーバ ー・ボールヴァード6830地
⑦発明者	リチャード・アレン・ ダイアン	アメリカ合衆国フロリダ州ボカ・ラトン、ノース・イース ト・73ストリート830番地
⑦発明者	スコット・ジェラル ド・キニアー	アメリカ合衆国フロリダ州ボカ・ラトン、サドルクリク・ ドライブ9005番地
⑦発明者	ジョージ・ディー・コ バツク	アメリカ合衆国フロリダ州ボカ・ラトン、ウエストブロッ ク・ドライブ19090番地
⑦発明者	マシュー・ステイブ ン・ボルカ、ジュニア	アメリカ合衆国ノース・カロライナ州ラレイ、プラス・ケ トル・ロード10800番地
⑦発明者	ロバート・サクセンマ イアー	アメリカ合衆国フロリダ州ボカ・ラトン、ノース・イース ト8番コート7329番地
⑦発明者	ケビン・マーシャル・ ジボロスキ	アメリカ合衆国ノース・カロライナ州・ラレイ、ウッドマ ナー・ドライブ・1313番地
⑦発明者	ジェリイ・デューン・ デキシオン	アメリカ合衆国フロリダ州ボカ・ラトン、エンフィールド ド・ストリート801番地
⑦発明者	アンドリユー・ボイ ス・マクネイル	アメリカ合衆国フロリダ州ディアフィールド・ビーチ、 ノース・ウエスト41番ウェイ181番地
⑦発明者	エドワード・イーブニ ング・ワツチテル	アメリカ合衆国フロリダ州ボカ・ラトン、ノース・イース ト・カイ・テラス500番地



**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☒ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☒ **FADED TEXT OR DRAWING**
- ☒ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**